HIPPI: Simplicity Yields Success

HIPPI hit the market at the right time with a simple high-speed interface that used very few options.

Don Tolmie and John Renwick

The High-Performance Parallel Interface (HIPPI) is a simplex point-to-point interface for transferring data at peak data rates of 800 or 1600 Mb/s over distances of up to 25 meters. A logical full-duplex circuit is accomplished by using two HIPPI interfaces.

HIPPI was first proposed to a working group of the ANSI Task Group X3T9.3 in January of 1987, and the first HIPPI implementations were available in late 1988. We feel that the HIPPI documents produced by X3T9.3 benefited the high-end computing community by producing the first world's first supercomputer interface standard. HIPPI was done quickly, especially when compared to other standards activities. A standard has a window of opportunity, and the developers must discipline themselves to cut off the design and stabilize the standard before the window closes.

Some of the committee goals that were developed during the early days include the following: provide a "fire hose" for moving data, keep it sweet and simple (KISS), get it done quickly, use options sparingly, and require no new silicon.

We feel that these goals were met, and that the goals were instrumental in the success of the project. It also should be noted that although the initial HIPPI proposal came from Los Alamos, it was greatly refined by the committee. Design by committee really worked; we aimed for a race horse and got a race horse, not a camel!

An early requirement for HIPPI was to support visualization at close to the 800 Mb/s peak data rate. For example, movies with 1K x 1K pixels, 24 bits of color per pixel, and 30 frames per second, require a continuous data rate of about 750 Mb/s. Anything that decreased this rate would show up on the screens, and could cause unacceptable behavior. This was our model of a "fire hose." We kept asking ourselves if an idea being proposed would make the transfers faster, or add overhead that could slow them down. If something is simple, it usually can be implemented in minimal hardware, which in turn can run at high speeds. When bells and whistles are added, the complexity grows and performance often suffers.

The KISS goal was invoked several times. A proposal would be made and developed over several months; mostly spent in figuring out how to do it. About the time everyone was accepting the proposal, someone would raise the question whether we should include it or not. As with most things, there is no free lunch. If you are going to add something, then it usually costs — and is the new feature worth the cost? It was a balancing act that depended on, and benefited from, the good judgment of the committee participants.

When the first proposals were made it was noted that it would be beneficial to use fiber optics to gain the benefits of longer distances and improved EMI/RFI characteristics. Unfortunately, we felt that the fiber technology then was not mature enough for our needs, and the desired time scale did not allow waiting. HIPPI hit the market at the right time, i.e., there were no standards for anything else in its class and vendors were just starting to consider higher speed channels. If we were going to meet the time goals of the participants, then it was going to have to be agreed upon quickly.

Many of the committee members felt that options were the antithesis of a standard. If there are lots of options, then the probability of multiple vendors using the same options, and hence interoperating, becomes smaller. The only option that survived in the physical layer was support for 800 or 1600 Mb/s. There are some options in the other layers, but they were kept to a minimum. Again, this is a balancing act; options add versatility, but at the cost of complexity.

The "no new silicon" goal also was driven by the desire to allow implementations to be completed soon. Early implementations, and even some current ones, were built with currently available small-scale integration parts. Today, HIPPI chip sets are available and should be considered for new designs.

The HIPPI document set is shown in Fig. 1, with each document limited in scope for ease of understanding. HIPPI-PH [1] defines the physical layer with the mechanical, electrical, and timing details. HIPPI-FP [2] defines the packet format, and header, for transferring large data blocks. HIPPI-LE [3] is a mapping to IEEE 802.2 so that communications protocols such as TCP/IP can use HIPPI. HIPPI-FC is a mapping to the fibre channel protocols. HIPPI-IPI is a mapping supporting high-performance storage systems using the IPI-3 disk and tape command sets. HIPPI-SC [4] defines the operation of HIPPI physical layer switches. In late 1992, HIPPI-PH and HIPPI-FCP were approved ANSI standards and the other documents were in various stages of development and approval. Serial-HIPPI [5] is an implementor's agreement, not a standard, defining a fiber-based HIPPI-PH extender for distances of up to ten kilometers.

The basic organization of the HIPPI informa-

DON TOLMIE is the chairman of ANSI Task Group X3T9.3. He is the initiator and leader of the HIPPI effort.

JOHN RENWICK is principal software designer at Netstar, Minneapolis, Minnesota.

0890-8044/92/$03.00 1993 ©IEEE

tion or data framing is shown in Fig. 2. A connection is made in a fashion similar to the connection made when dialing the telephone. Once a connection is established, a packet (or multiple packets) can be sent from the source to the destination. Each packet contains one or more bursts, and each burst contains one to 256 words. Bursts that contain less than 256 words may occur only as the first or last burst of a packet. Words are composed of 32 or 64 bits with odd parity on each byte.

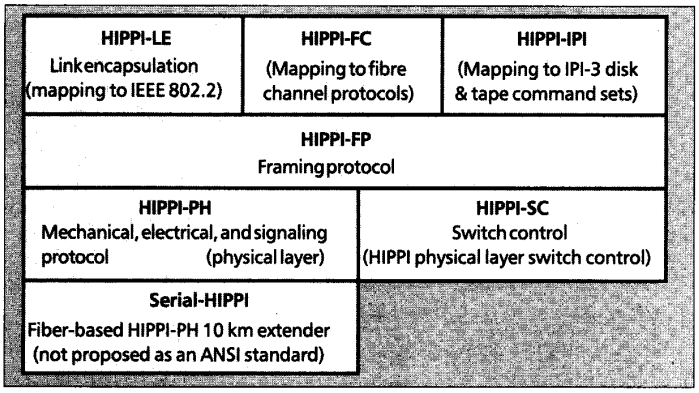
Physical Layer — HIPPI-PH

HIPPI-PH specifies 50-pair, twisted-pair cables for distances up to 25 meters. The 800-Mb/s (32-bit) option uses one cable, and the 1600-Mb/s (64-bit) option uses two cables. The HIPPI signal lines are unidirectional to accommodate fiber-optic implementations and crossbar switches. The control and data signals use differential emitter-coupled logic (ECL) drivers and receivers, and are timed in relation to the continuous 25-MHz CLOCK signal. The signal set includes:

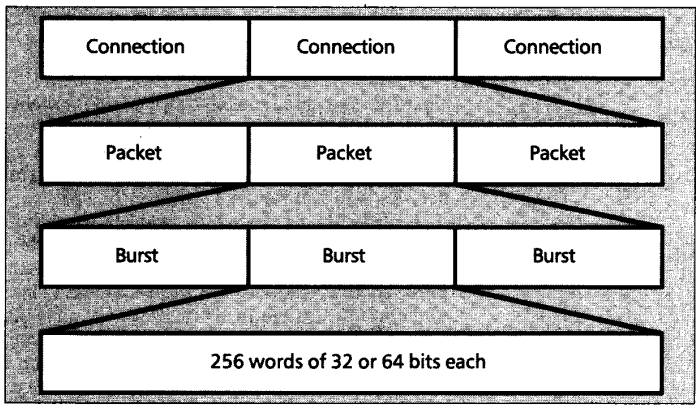
REQUEST: source asks to establish a connection
CONNECT: destination accepts the connection
READY: gives the source permission to send a burst
PACKET: brackets one or more bursts into a packet
BURST: brackets 256 data words on contiguous clocks
DATA BUS: 32 or 64 bits (800 or 1600 Mb/s)
PARITY BUS: 4 or 8 bits for DATA BUS odd byte parity
CLOCK: continuous 25 MHz, 40 nanosecond period
INTERCONNECT: cables connected and power ON.

Typical HIPPI wave forms are shown in Fig. 3 for a sequence that establishes a connection, sends a packet containing two bursts, sends a packet containing one burst, and then disconnects. (S) after a signal name means that the source drives the signal, and (D) means that the destination drives the signal. A connection is made from the source to the destination much like a telephone connection. The source supplies the I-Field on the data bus (like a telephone number), and asserts the REQUEST signal. If the destination can accept the request, it asserts the CONNECT signal completing the connection. Once a connection is established, single or multiple packets may be transferred from the source to the destination. Packets are delimited by the PACKET signal being true. Packets are composed of one or more bursts. The LLRC is a one-word, even-parity checksum for each burst. Either the source or destination can break the connection by dropping the REQUEST or CONNECT signal, respectively.

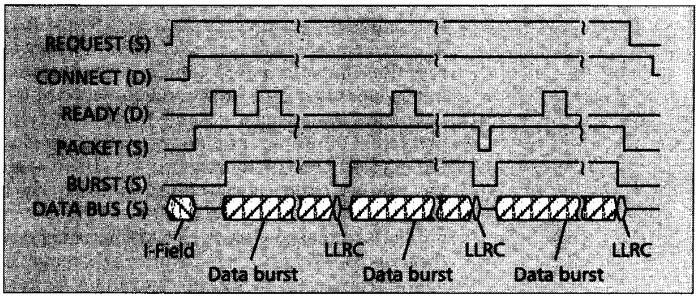
One of the first things that was decided for HIPPI was that it should be a point-to-point interface rather than a multi-drop cable. This simplified the electrical design considerably by eliminating wiring stubs, variable loads, and complex reflection problems. Copper twisted-pair cables were selected when a cable vendor provided a low-skew cable with excellent shielding characteristics. The 25-meter maximum cable length was chosen to be what was easily achieved rather than pushing the limit and having to tweak implementations. Also, the HIPPI components were based on existing commercial components rather than pushing the state of the art and depending upon something that may not become available within our time frame goal.



■ **Figure 1.** *HIPPI documents.*



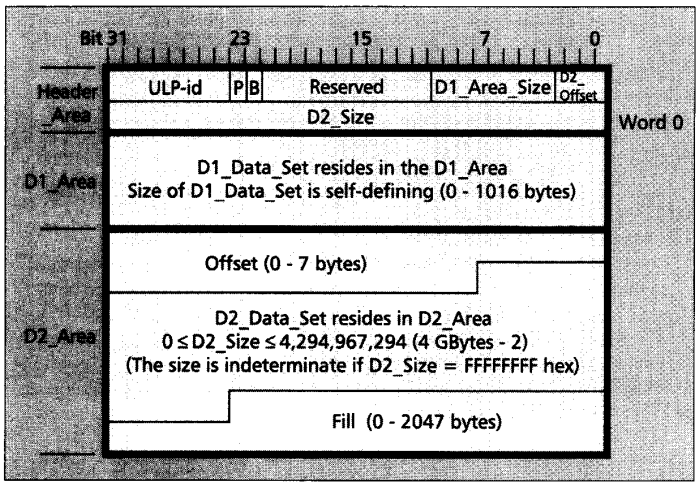
■ **Figure 2.** *HIPPI data framing.*



■ **Figure 3.** *Typical HIPPI wave forms.*

The HIPPI flow control was designed to accommodate the longer distances afforded by future fiber-optic-based systems. A HIPPI destination generates a READY signal to give the source permission to send a burst of up to 256 words (1024 or 2048 bytes). The destination can issue multiple READY signals according to its current buffering capability. These READY signals are queued by the source so that when data transmission is desired, round trip handshake delays do not occur. This scheme is similar to a window in a higher layer protocol. Supporting the full 800 Mb/s bandwidth requires about one kilobyte of buffering at the receiver for each kilometer of distance.

There were some requests for additions to HIPPI to support real-time peripherals, and these additions



■ **Figure 4.** *HIPPI packet format.*

would have required some additional HIPPI signals. The committee's decision was to limit the scope of HIPPI to the memory-to-memory model rather than trying to make it all things to all people. This decision also emphasizes the KISS principle.

HIPPI-PH specifies simple byte and burst (horizontal and vertical) parity-based error detection. Error correction is done by upper-layer protocol-initiated retransmissions. This parity-based error detection design results in detection of all 1-bit, 2-bit, 3-bit, odd-bit, and greater than 99.99 percent of 4-bit errors within a burst. A proposal to add a 64-bit cyclic redundancy check (CRC) was made late in the development cycle. Although this would have made HIPPI's data integrity completely unassailable, the committee rejected the change because: 1) it required a new chip that was not yet available; 2) HIPPI appeared already to have such a low error rate that the anticipated gain in integrity could not be measured; 3) it would require hardware redesign and would make existing implementations obsolete; 4) It would send an unintended signal to the implementer community that the HIPPI specification was seriously flawed and unstable, which could result in the death of HIPPI.

The last point may have weighed heaviest against the proposal. When a design has passed thorough review, multiple implementations, and testing, the standards body must foster a sense of confidence among implementers that the design is indeed adequate, complete, and stable. If the engineers don't stop making improvements, they may never see the product.

Framing Protocol — HIPPI-FP

HIPPI-FP specifies the packet format and header for transferring data blocks across HIPPI interfaces (Fig. 4). HIPPI-FP specifies two data areas, called D1 and D2, within the packet. D1 is intended for control information, is limited to 1016 bytes, and is contained within the first burst of a packet (P=1 if D1 is present). D2 is intended for the user data, and can be exactly specified up to 4 gigabytes -2, or left indeterminate for infinite data streams or unknown sizes. The intent of the separate D1 and D2 data sets is that the D1 set is passed to the upper layer protocol when it is received so that this control information may be parsed while the D2 user data is still in transit.

Another feature of HIPPI-FP is an offset value of up to 7 bytes for the D2 data set. This was included to avoid having to byte align the data if it was not on a natural word boundary. HIPPI-FP also includes the ULP-id field identifying the upper layer protocol to which the packet belongs. The B bit in the header indicates if the D2 data set starts in the first burst, or at the start of the second burst. Starting D2 on a burst boundary allows an easy split of the control and data information into separate memory buffers.

Link Encapsulation — HIPPI-LE

HIPPI-LE specifies a header format for encapsulating IEEE 802.2 logical link control protocol data units for transmission over HIPPI. This allows many communications protocols, such as TCP/IP, to be used with HIPPI. Included in the HIPPI-LE header are source and destination 48-bit IEEE Universal LAN MAC addresses. Some vendor unique fields also are included for potential use as short addresses in local networks. The HIPPI-LE header is placed in the HIPPI-FP D1 data set, and the IEEE 802.2 and other information is placed in the HIPPI-FP D2 data set.

Mapping to Fibre Channel — HIPPI-FC

HIPPI-FC will specify a method for interconnecting HIPPI-based equipment and fibre channel equipment. At the time this was written this project was in a very early definition and development phase in X3T9.3. A companion project, FC-FP, is intended to provide a mapping to allow HIPPI upper-layer protocols to use the fibre channel physical layer transmission system. The HIPPI-FC project will map at the physical layer rather than the upper-layer protocol.

Mapping to IPI-3 — HIPPI-IPI

This specification is used to transport the IPI-3 Disk and IPI-3 Tape commands over HIPPI. It is being used today to support high-performance storage systems attached through HIPPI. The committee presently plans to put the information being generated for this document in revisions of the IPI-3 Disk and IPI-3 Tape documents, and not have a separate HIPPI-IPI document.

It was work on using the IPI-3 command sets that led to the split of the D1 and D2 data sets in HIPPI-FP. This allows the destination to be parsing the IPI-3 command information while the user data is still in transit.

Switch Control — HIPPI-SC

The HIPPI physical layer, HIPPI-PH, specifies a point-to-point link, and interconnecting only two devices is not very interesting in most installations. To obtain the equivalent of multi-drop capability, HIPPI physical layer switches can be used to interconnect many devices in a local area network. Crossbar switches are the most common interconnection mechanism used with HIPPI. The use and control of these crossbar switches is specified in HIPPI-SC.

Fiber Extender — Serial-HIPPI

Serial-HIPPI is an implementor's agreement defining a transparent fiber-optic link for distances up to 10 kilometers. A 20b/24b transmission code is used to provide DC balance for the serial signal, and to provide bit, byte, and word synchronization. Distances up to 36 meters are also possible with copper coaxial cable. The serial stream operates at a 1200 Mb/s rate with the 800 Mb/s HIPPI-PH, the difference between the rates being the overhead of the HIPPI parity and control signals and the 20b/24b coding. An integrated circuit chipset, which translates between a 20-bit parallel stream and the serial stream, is available. Complete outboard extenders also are available to convert a parallel electrical HIPPI-PH interface into Serial-HIPPI and back. Serial-HIPPI uses one fiber in each direction for a bidirectional pair of 800 Mb/s HIPPI interfaces.

HIPPI Tester

We feel that one of the reasons that HIPPI has been successful is that a HIPPI-PH tester has been available for systems designers. The tester was designed and prototyped at Los Alamos to test the initial systems built at Los Alamos, and those provided by vendors for Los Alamos use. After the first few HIPPI Tester units were built and proven, the design was transferred to a commercial company for quantity production and support. The HIPPI Tester has proven to be very useful in 1) initial design checkout, 2) implementation testing before shipping, and 3) field testing of suspect HIPPI interfaces. It is packaged in a briefcase for ease of use and transport, including a laptop PC used for control.

This inexpensive unit provides a set of test suites that allow an engineer to verify an implementation against a known set of tests. The HIPPI Tester can spot illegal signal transitions and data errors and record the conditions when they occur. Data patterns can also be programmed, and transmitted to host computers, or looped and checked in the tester. The tester has allowed vendors to debug their new systems in the comfort of their own laboratories, with reasonably good assurance that the finished product would interoperate with other vendor's equipment in the field. The resultant "plug and play" has proven very cost effective and also is a major time saver when first connecting systems from different vendors.

How Do We Measure Success?

A measure of success is that many vendors have built fully compliant HIPPI-PH interfaces without hand-holding. This hand-holding includes attending the standards meetings, calling the document author (one of the authors of this paper), or participating in HIPPI training sessions. This leads us to the conclusion that the documents are understandable, and the HIPPI concepts are reasonable to implement.

HIPPI's Role in Networks

HIPPI is a data channel, not a network medium. However it has important networking roles to play. It is one of several available interfaces to Ultra Network Technologies' proprietary UltraNet, which links supercomputers and workstations on gigabit-per-second back planes. HIPPI-SC switches can connect a number of computer systems together as a local area network. At least one ANSI fibre channel fabric product is being developed with HIPPI interfaces.

HIPPI's unique potential as a network interface comes from two of its characteristics: its speed (it is the only widely available standard interface fast enough for use with emerging gigabit networks) and its connection-oriented hardware protocol. HIPPI was designed with crossbar switches in mind.

The crossbar switch makes an interesting LAN because it has an aggregate bandwidth that is a multiple of the peak data rate of any single interface. Multiple simultaneous connections can exist through a switch, and since connections share no switch resources, they can pass data concurrently at the full HIPPI rate. This is a desirable characteristic for distributed applications and distributed storage, in which many simultaneous interactions may have to take place.

Available HIPPI-SC switches allow up to 32 hosts to be interconnected on a single hub, and hubs can be interconnected for larger, more distributed networks. Connection switching times are typically less than one microsecond, making the HIPPI-SC switch practical as a connection-oriented network medium. Work is going on within the IETF on an Internet Draft, describing the host interface to such networks, based on the HIPPI-SC and HIPPI-LE draft standards. This draft standard, RFC 1374, "IP and ARP on HIPPI" draft was submitted by J. Renwick and A. Nicholson of Cray Research, Inc. in June 1992. Several implementers demonstrated interoperable TCP/IP on a HIPPI-SC-based LAN at the Supercomputing '92 conference.

A HIPPI crossbar switch is an interconnection matrix with HIPPI interfaces. Each simplex HIPPI coming into the switch can be connected to one going out. The switch connection is made electrically at the time the HIPPI connection is negotiated using the REQUEST and CONNECT signals; the switch uses the I-Field data to determine which output port is being requested. If the requested destination already is connected to another source, the switch can respond in two ways according to HIPPI-SC: it can immediately reject the new request by asserting the CONNECT signal for a brief interval, or it can delay asserting CONNECT until the existing connection to the requested destination is released (this is called "camp-on"). A bit in the requester's I-Field selects the desired action. The requester can time out and withdraw an unsatisfied camp-on request by dropping its REQUEST signal. It may then try a different destination by reasserting REQUEST with a different I-Field value.

HIPPI-SC describes two modes of switch addressing. One is called "Source Route" mode. In this mode, the I-Field contains a 24-bit string of port numbers which give the explicit route through a series of cascaded switches. The other mode, called "Logical Address" mode, is preferred for networking. A logical address is a flat 12-bit address for a destination which is the same for all sources, regardless of how many links must be traversed between originating and destination switches. Logical ad-

■ ■ ■ ■ ■

One of the first things that was decided for HIPPI was that it should be a point-to-point interface rather than a multi-drop cable.

The success of HIPPI for networking will depend on the availability of Serial-HIPPI products which end the need for parallel copper cables.

dressing practically requires that each switch have look-up tables to map from 12-bit addresses to physical ports.

According to the current proposal for IP on HIPPI, hosts will use connections of very short duration, only long enough to send a limited number of packets that can be sent without delay. Maximum IP packet size is about 64K bytes. If each host can keep the overhead of setting up connections very short (i.e., within 10 or 20 μs), very high HIPPI utilization is possible even though only one packet is sent per connection.

Cray Research, Inc., operates a HIPPI-SC LAN consisting of two switches and a handful of attached supercomputers. Hosts on this network send only one packet per connection, use camped-on connections exclusively, and transmit packets strictly in the order queued, regardless of destination. Point-to-point throughput of up to 75 megabytes per second can be shown with maximum-sized TCP/IP packets. Since each packet is nearly as big as the entire window in a standard TCP connection, this speed is possible only if a TCP window scale factor (RFC 1323 [6]) is used.

HIPPI has the disadvantage of not naturally supporting broadcast, a feature of conventional LANs upon which some applications and protocols depend, in particular the Internet Address Resolution Protocol. (ARP discovers the physical, or hardware, address of a host whose Internet address is known. Conventionally ARP works by broadcasting a request to all hosts on the LAN; the host whose Internet address is contained in the request responds with a reply containing its hardware address.)

ARP resolves Internet addresses to 48-bit Ethernet (IEEE) addresses, but HIPPI switches do not use IEEE addresses; they require physical port addresses that are local to the switch domain. The previously described "IP and ARP on HIPPI" Internet Draft proposes that Internet ARP be piggybacked on a similar address resolution protocol within HIPPI-LE, which maps IEEE addresses to the switches' logical addresses. These piggybacked packets are sent on a connection to a logical address reserved for multicast.

True multicast could in principle be done by the switch itself by connecting a source to multiple destinations at the same time and sending data to all simultaneously, at a rate paced by the slowest destination. The multicast could not take place until all destinations were available for the connection. This mechanism has never been implemented; a more likely solution is a server attached to the switch, which receives a multicast message, stores it temporarily and forwards it to a list of destinations one by one.

Alternatively, ARP packets could be intercepted and replied to by an address server acting on behalf of the hosts whose addresses are requested. Either method may be desirable depending on cost, security, administration methods or other considerations; the hosts themselves can be designed to work equally well with either.

Serial-HIPPI and Networking

HIPPI got a head start in high-speed networking because it used parallel copper cable at a time when serial encoder/decoder chips and fiber optic technology was unavailable or prohibitively priced at gigabaud rates. While useful in limited size networks within computer centers, HIPPI cables are too short, fat, and cumbersome for LANs in general. The good news is that a proprietary fiber optic HIPPI extender from Broadband Communication Products, Inc., has already demonstrated complete transparency and reliability on a par with copper HIPPI over a distance of 10 kilometers. During 1992 at least two vendors are expected to show similar performance with interoperable Serial-HIPPI which conforms to the implementors' agreement.

Serial-HIPPI will be implemented first as HIPPI "modems," converting parallel copper to fiber optic and back. The next step will be to place the serial HIPPI chipset and transceiver on the board, bypassing the parallel transceivers for a native serial channel. Integrated Serial-HIPPI will permit a big jump in the size of switch hubs, which today are limited in size by the number and density of HIPPI bulkhead connectors and by the number of parallel paths that must be interconnected. The success and widespread use of HIPPI for networking will depend substantially on the availability of Serial-HIPPI products which end the need for parallel copper cables.

Acknowledgments

The Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36. This work was performed under auspices of the U.S. Department of Energy. This paper was assigned number LA-UR 92-1578.

References

[1] ANSI X3.183-1991, "High-Performance Parallel Interface — Mechanical, Electrical, and Signalling Protocol Specification (HIPPI-PH)."

[2] ANSI X3.210-199x, "High-Performance Parallel Interface — Framing Protocol, (HIPPI-FP)."

[3] ANSI X3.218-199x, "High Performance Parallel Interface — Encapsulation of ISO 8802-2 (IEEE Std 802.2) Logical Control Protocol Data Units (HIPPI-LE)."

[4] ANSI X3.222-199x, "High-Performance Parallel Interface — Physical Switch Control (HIPPI-SC)."

[5] "Serial-HIPPI Specification, Revision 1.0, Serial-HIPPI Implementers Group."

[6] V. Jacobson, "TCP Extensions for High Performance," RFC 1323.

Biography

DON TOLMIE (M '59) received a B.S.E.E. degree from New Mexico State University in 1959 and an M.S.E.E. degree from University of California, Berkeley, in 1961. He joined the Los Alamos National Laboratory in 1959 as a technical staff member, and has been involved with networking of supercomputers for almost 20 years. Currently he is the chairman of ANSI Task Group X3T9.3. He is the initiator and leader of the HIPPI effort.

JOHN RENWICK received a BA in Mathematics from Michigan State University in 1975. He was employed there as a programmer in operating systems and networks until he joined Cray Research in 1981 as a senior programmer analyst responsible for networking aspects of HIPPI. He has represented Cray Research on the ANSI Task Group X3T9.3 since 1988 and participated actively in the development of the HIPPI standard. He currently is a principal software designer at Netstar